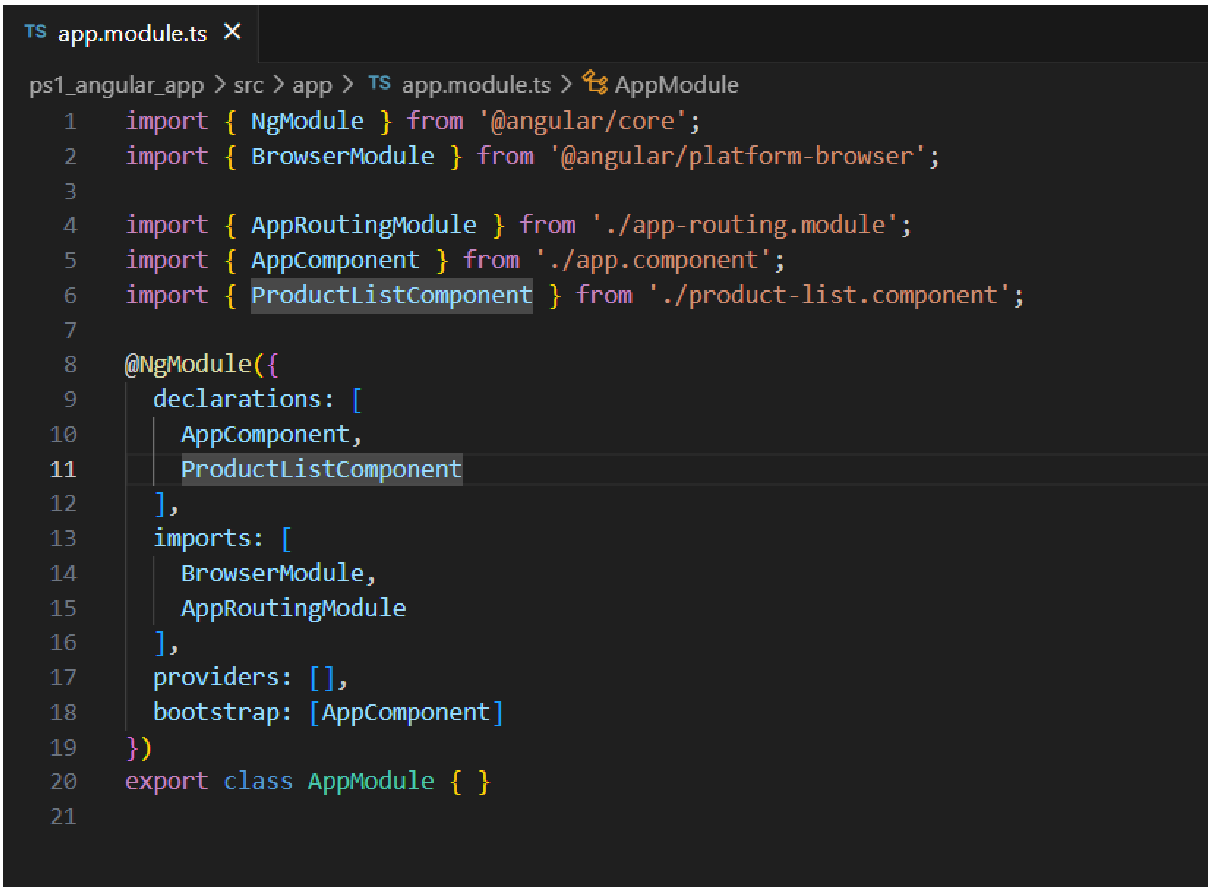
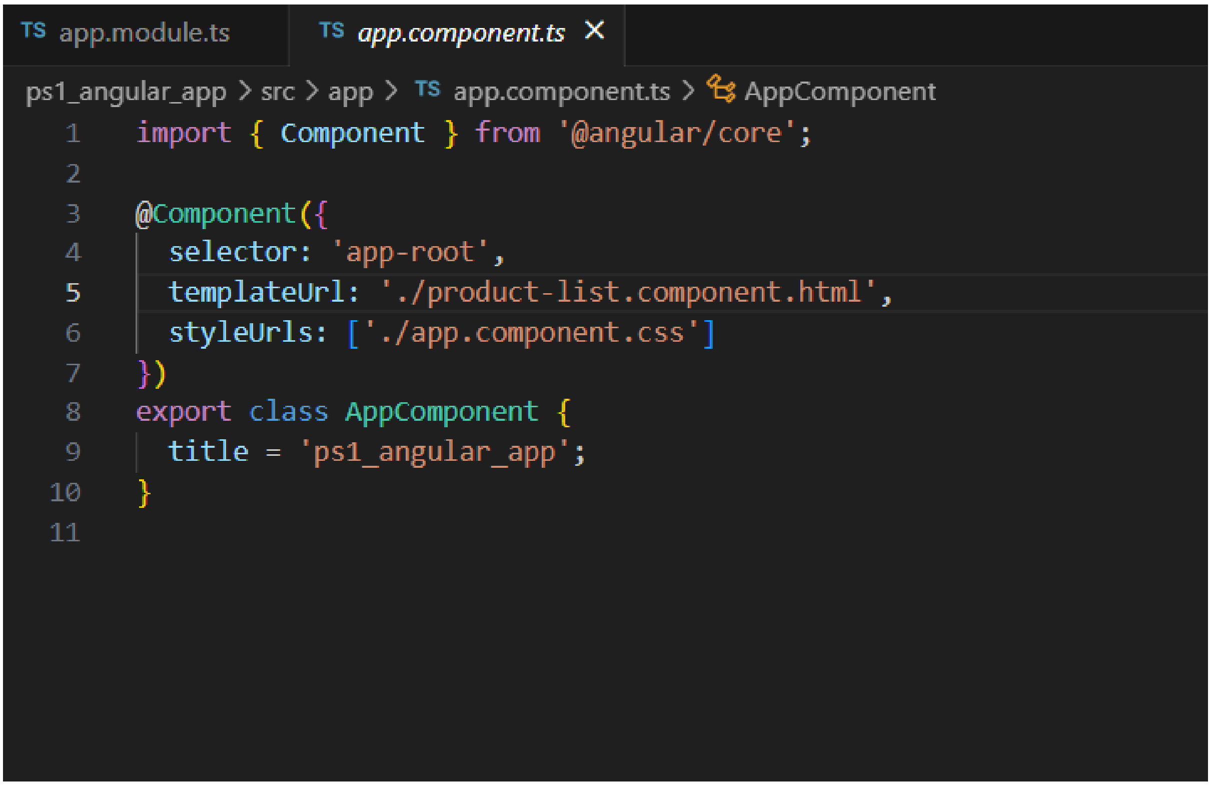
Problem-Statement 1



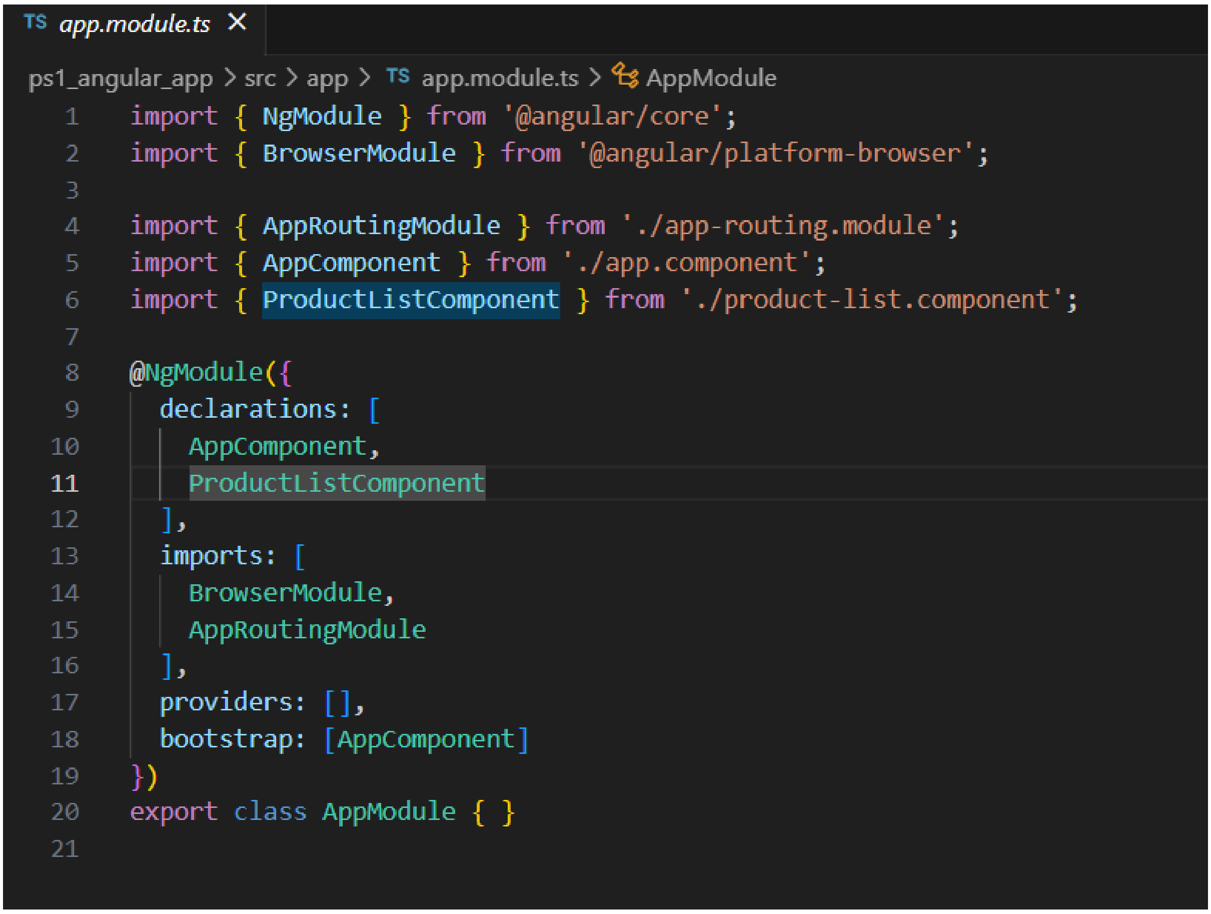
```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { ProductListComponent } from './product-list.component';

@NgModule({
  declarations: [
    AppComponent,
    ProductListComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```


TS app.module.ts
TS app.component.ts ✕
ps1_angular_app > src > app > TS app.component.ts > AppComponent

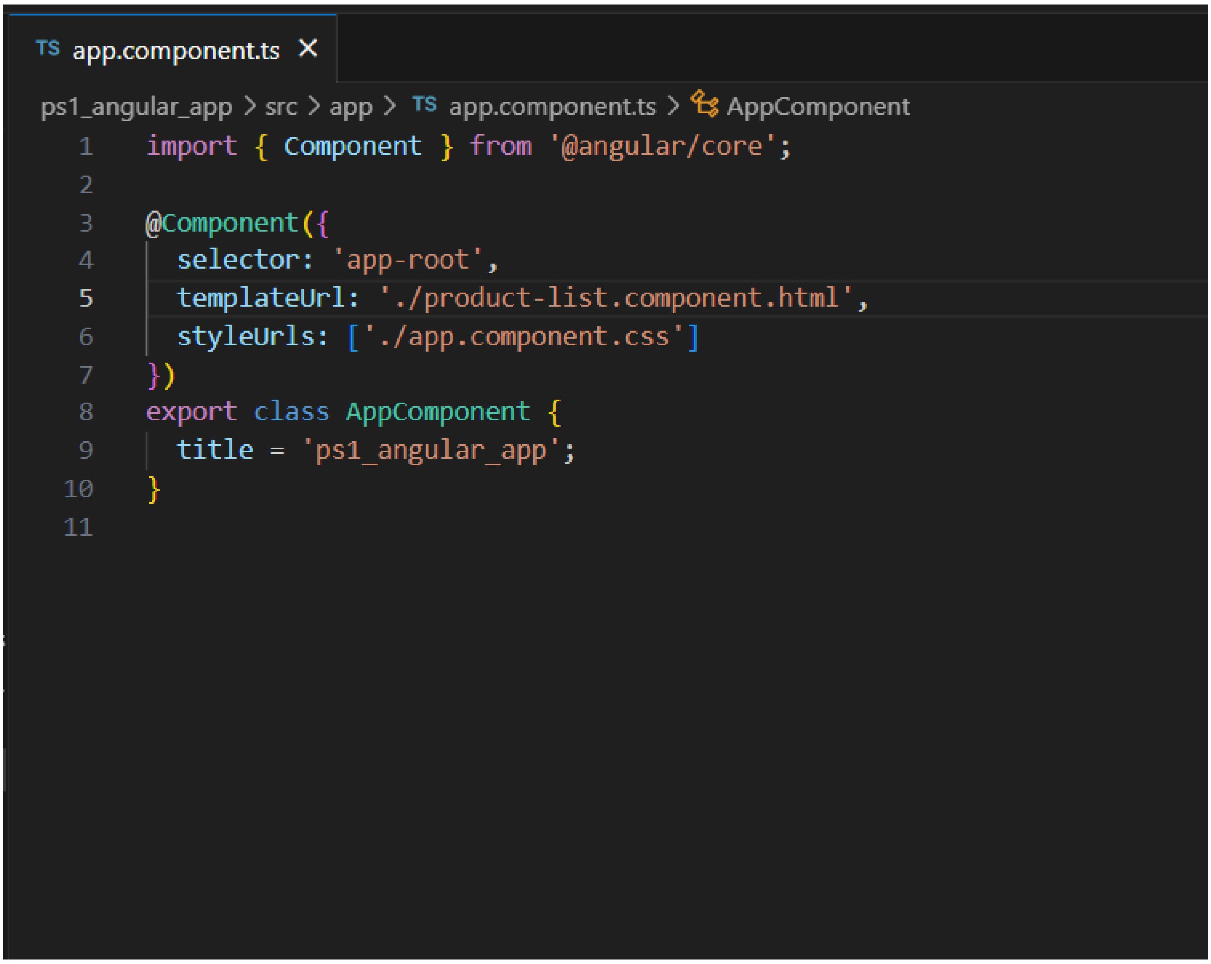
```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './product-list.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'ps1_angular_app';
}
```

Product-List

Problem-Statement 2



```typescript
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { ProductListComponent } from './product-list.component';

@NgModule({
  declarations: [
    AppComponent,
    ProductListComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

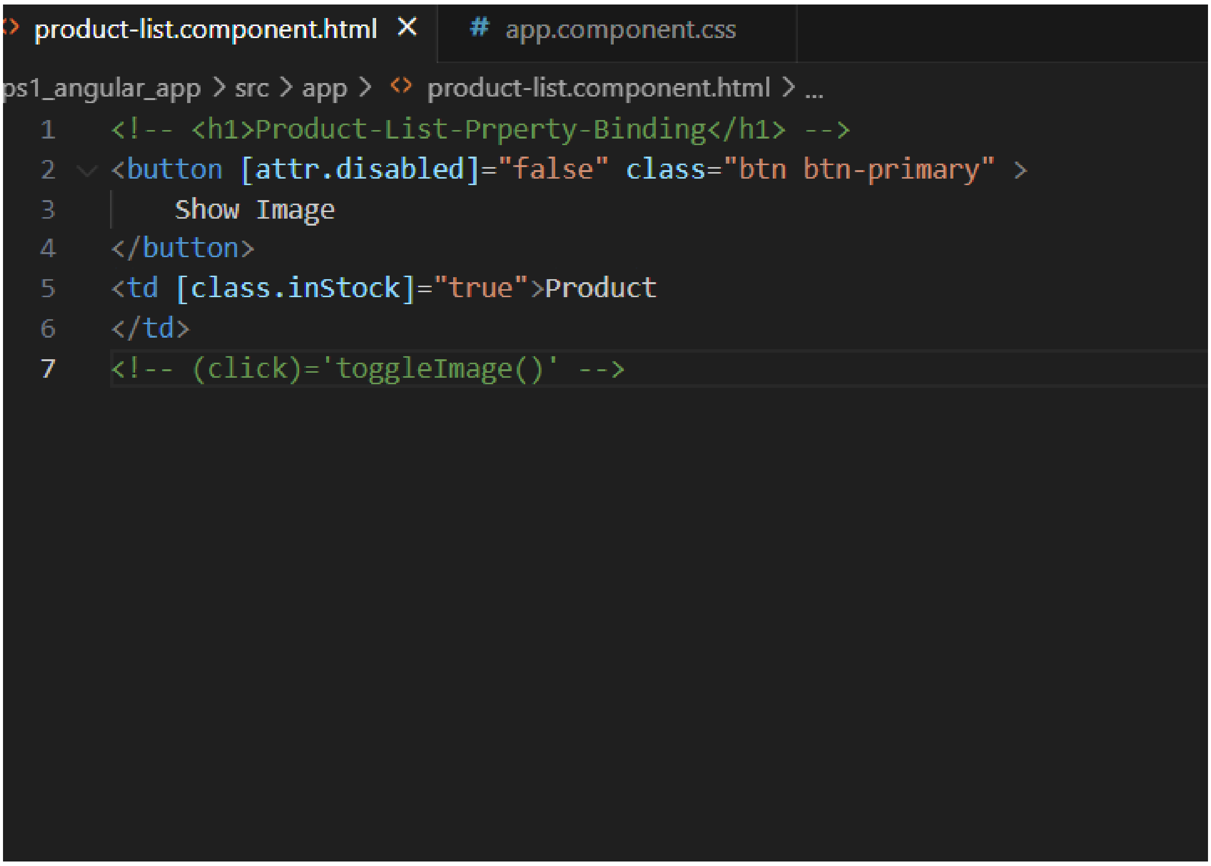


TS app.component.ts ✕
ps1_angular_app > src > app > TS app.component.ts > AppComponent

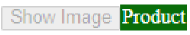
```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './product-list.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'ps1_angular_app';
}

```

Product-List-Prperty-Binding

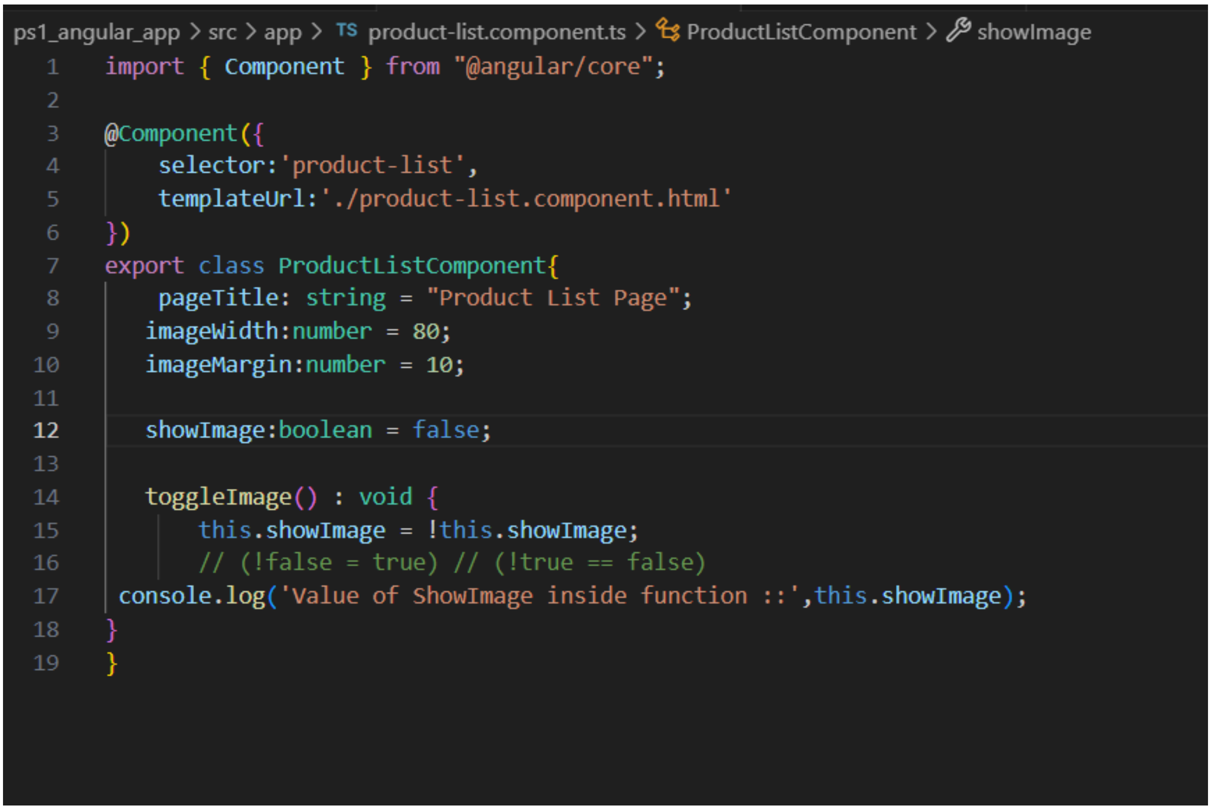
Problem-Statement 3


product-list.component.html
app.component.css
ps1_angular_app > src > app > product-list.component.html > ...

```
<!-- <h1>Product-List-Prperty-Binding</h1> -->
<button [attr.disabled]="false" class="btn btn-primary" >
    Show Image
</button>
<td [class.inStock]="true">Product
</td>
<!-- (click)='toggleImage()' -->
```



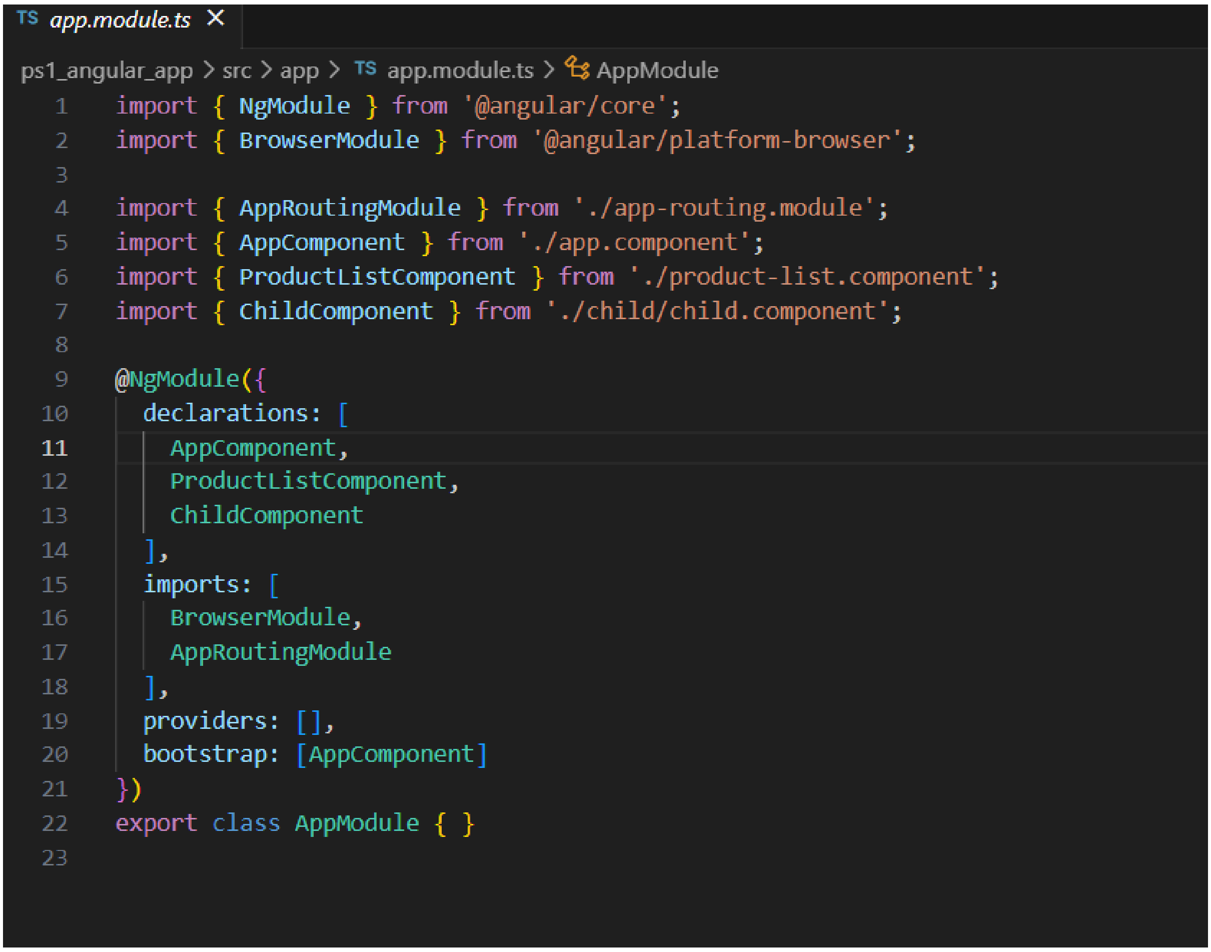
<u>Problem-Statement 4</u>



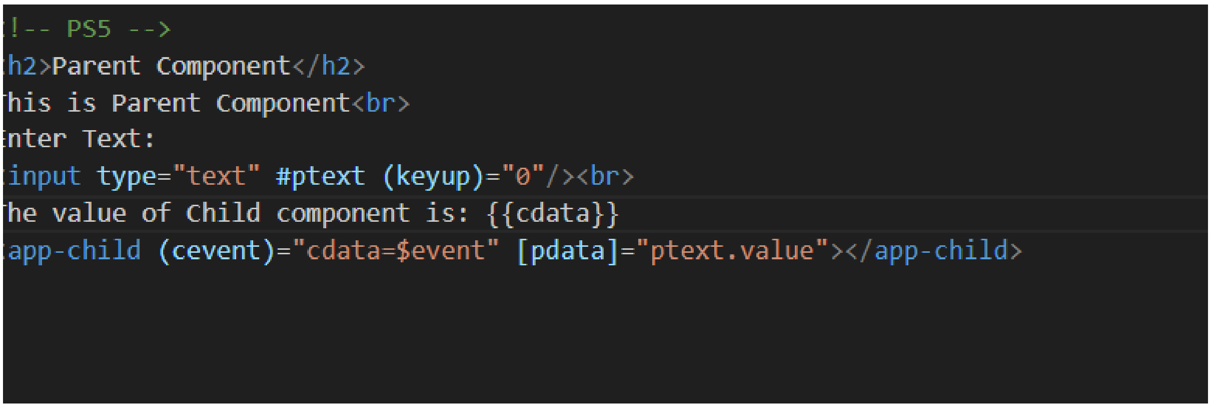
ps1_angular_app > src > app > TS product-list.component.ts > ProductListComponent > showImage

```
import { Component } from "@angular/core";

@Component({
    selector:'product-list',
    templateUrl:'./product-list.component.html'
})
export class ProductListComponent{
    pageTitle: string = "Product List Page";
   imageWidth:number = 80;
   imageMargin:number = 10;

   showImage:boolean = false;

   toggleImage() : void {
       this.showImage = !this.showImage;
       // (!false = true) // (!true == false)
 console.log('Value of ShowImage inside function ::',this.showImage);
}
}
```

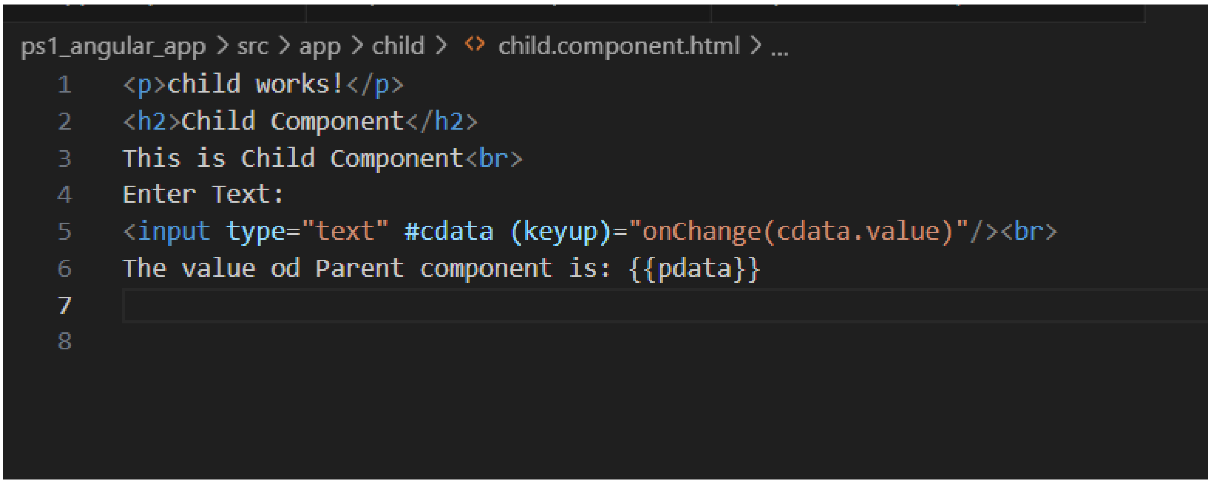
Problem-Statement 5



```typescript
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { ProductListComponent } from './product-list.component';
import { ChildComponent } from './child/child.component';

@NgModule({
  declarations: [
    AppComponent,
    ProductListComponent,
    ChildComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

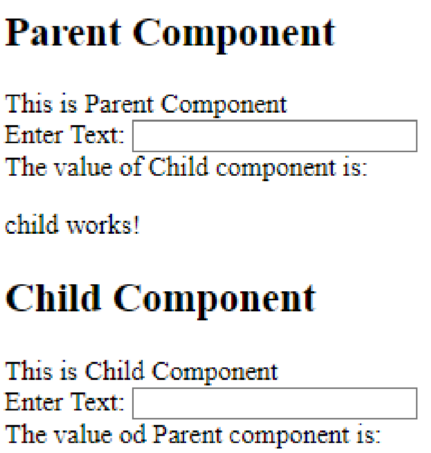


```html
!-- PS5 -->
h2>Parent Component</h2>
his is Parent Component<br>
nter Text:
input type="text" #ptext (keyup)="0"/><br>
he value of Child component is: {{cdata}}
app-child (cevent)="cdata=$event" [pdata]="ptext.value"></app-child>
```



ps1_angular_app > src > app > child > child.component.html > ...

```html
<p>child works!</p>
<h2>Child Component</h2>
This is Child Component<br>
Enter Text:
<input type="text" #cdata (keyup)="onChange(cdata.value)"/><br>
The value od Parent component is: {{pdata}}

```



Parent Component

This is Parent Component
Enter Text:
The value of Child component is:

child works!

Child Component

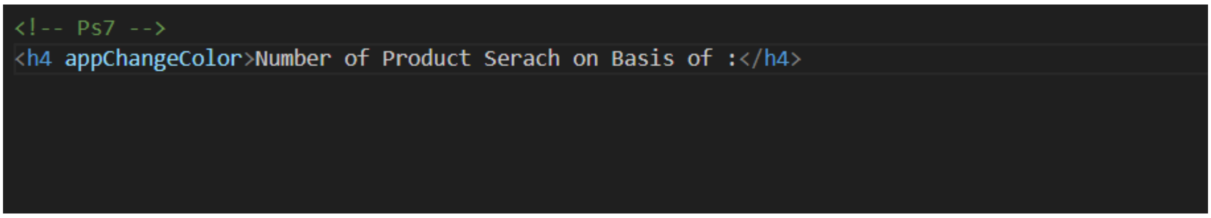
This is Child Component
Enter Text:
The value od Parent component is:

Problem-Statement 6

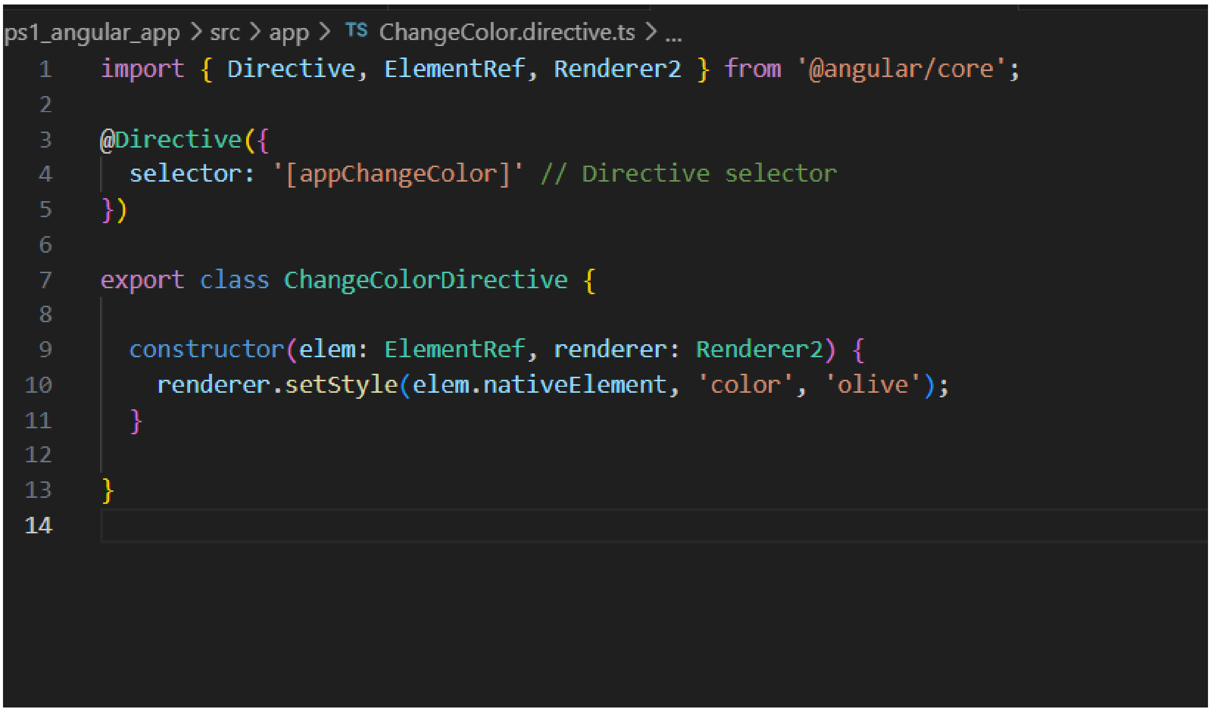
```html
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Page Title</title>
```

```
</head>
<body style="padding:40px ">
<p style="text-align: center;font-size: 30px">  Hello <input
type="text" placeholder="Your name" (input)="ontyping($event)"/>,
Welcome to the <b>angCare!</b></p>
<p style="text-align: center;font-size: 20px">Hello {{name}}<br>I am
Joe, your personal assistant! I will guide you further...</p>
<div style="text-align:left;padding: 20px">
<p> Click on Sign up to create your account with angCare: <button
(click)="signup()" class="btn btn-primary">Sign up with
{{title}}</button>   : {{status}}</p>
<p> Click on Sign up to create your account with angCare: <button>Sign
In </button></p>
</div>
<div >
<div class="jumbotron">
  <div class="container">
      <div class="row">
          <div class="col-md-6 offset-md-3">
              <h2>Angular 6 Reactive Form Validation</h2>
              <form [formGroup]="registerForm" (ngSubmit)="onSubmit()">
                  <div class="form-group">
                      <label>First Name</label>
                      <input type="text" formControlName="firstName"
class="form-control" [ngClass]="{ 'is-invalid': submitted &&
f.firstName.errors }" />
                      <div *ngIf="submitted && f.firstName.errors"
class="invalid-feedback">
                          <div
*ngIf="f.firstName.errors.required">First Name is required</div>
                      </div>
                  </div>

                  <div class="form-group">
                      <label>Last Name</label>
                      <input type="text" formControlName="lastName"
class="form-control" [ngClass]="{ 'is-invalid': submitted &&
f.lastName.errors }" />
                      <div *ngIf="submitted && f.lastName.errors"
class="invalid-feedback">
                          <div *ngIf="f.lastName.errors.required">Last
Name is required</div>
                      </div>
                  </div>
                  <div class="form-group">
                      <label>Email</label>
                      <input type="text" formControlName="email" class="form-control" [ngClass]="{ 'is-invalid': submitted && f.email.errors }" />
                      <div *ngIf="submitted && f.email.errors" class="invalid-feedback">
                          <div *ngIf="f.email.errors.required">Email is required</div>
                          <div *ngIf="f.email.errors.email">Email must be a valid email address</div>
                      </div>
                  </div>
                  <div class="form-group">
                      <label>Password</label>
                      <input type="password" formControlName="password" class="form-control" [ngClass]="{ 'is-invalid': submitted && f.password.errors }" />
                      <div *ngIf="submitted && f.password.errors" class="invalid-feedback">
                          <div *ngIf="f.password.errors.required">Password is required</div>
                          <div *ngIf="f.password.errors.minlength">Password must be at least 6 characters</div>
                      </div>
                  </div>
                  <div class="form-group">
                      <button [disabled]="loading" class="btn btn-primary">Register</button>
                  </div>
              </form>
          </div>
      </div>
  </div>
</div>
</div>
</body>
</html>
<router-outlet></router-outlet>
```

Problem-Statement 7



```html
<!-- Ps7 -->
<h4 appChangeColor>Number of Product Serach on Basis of :</h4>
```

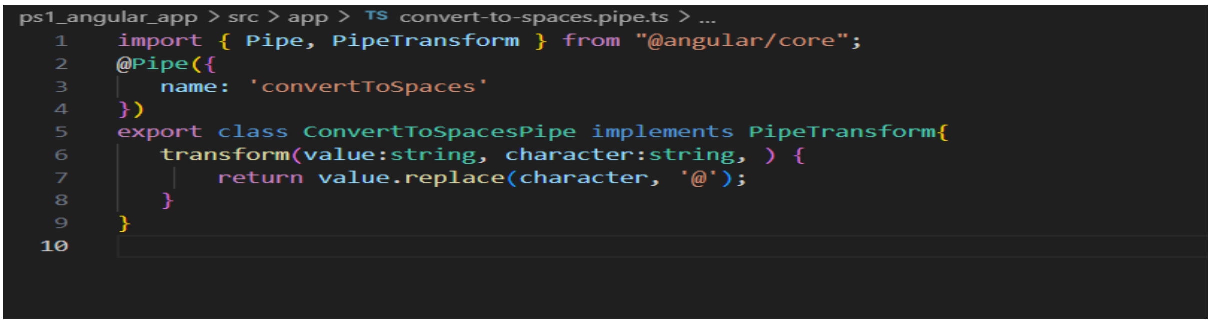


ps1_angular_app > src > app > TS ChangeColor.directive.ts > ...

```typescript
import { Directive, ElementRef, Renderer2 } from '@angular/core';

@Directive({
  selector: '[appChangeColor]' // Directive selector
})

export class ChangeColorDirective {

  constructor(elem: ElementRef, renderer: Renderer2) {
    renderer.setStyle(elem.nativeElement, 'color', 'olive');
  }

}

```

Number of Product Serach on Basis of :

Problem-Statement 8



ps1_angular_app > src > app > TS convert-to-spaces.pipe.ts > ...

```typescript
import { Pipe, PipeTransform } from "@angular/core";
@Pipe({
    name: 'convertToSpaces'
})
export class ConvertToSpacesPipe implements PipeTransform{
    transform(value:string, character:string, ) {
        return value.replace(character, '@');
    }
}

```

Uppercase

Problem-Statement 9

```
<p>
  signup works!
</p>

<div class="jumbotron">
  <div class="container">
        <div class="row">
        <div class="col-md-6 offset-md-3">
        <h2>Angular 6 Reactive Form Validation</h2>
        <form [formGroup]="registerForm" (ngSubmit)="onSubmit()">
                <div class="form-group">
                <label>First Name</label>
                <input type="text" formControlName="firstName"
class="form-control" [ngClass]="{ 'is-invalid': submitted && f.firstName.errors
}" />
                <div *ngIf="submitted && f.firstName.errors"
class="invalid-feedback">
                     <div *ngIf="f.firstName.errors.required">First Name is
required</div>
                </div>
                </div>
```

```html

        <div class="form-group">
        <label>Last Name</label>
        <input type="text" formControlName="lastName" class="form-control" [ngClass]="{ 'is-invalid': submitted && f.lastName.errors }" />
        <div *ngIf="submitted && f.lastName.errors" class="invalid-feedback">
            <div *ngIf="f.lastName.errors.required">Last Name is required</div>
        </div>
        </div>
        <div class="form-group">
        <label>Email</label>
        <input type="text" formControlName="email" class="form-control" [ngClass]="{ 'is-invalid': submitted && f.email.errors }" />
        <div *ngIf="submitted && f.email.errors" class="invalid-feedback">
            <div *ngIf="f.email.errors.required">Email is required</div>
            <div *ngIf="f.email.errors.email">Email must be a valid email address</div>
        </div>
        </div>
        <div class="form-group">
        <label>Password</label>
        <input type="password" formControlName="password" class="form-control" [ngClass]="{ 'is-invalid': submitted && f.password.errors }" />
        <div *ngIf="submitted && f.password.errors" class="invalid-feedback">
```

```
                              <div *ngIf="f.password.errors.required">Password is required</div>

                              <div *ngIf="f.password.errors.minlength">Password must be at least 6 characters</div>

                    </div>

                    </div>

                    <div class="form-group">

                    <button [disabled]="loading" class="btn btn-primary">Register</button>

                    </div>

          </form>

          </div>

          </div>

  </div>

</div>



<router-outlet></router-outlet>
```

```
import { Component, OnInit } from '@angular/core';

import {FormBuilder, FormGroup, Validators } from '@angular/forms';

@Component({

  selector: 'app-signup',

  templateUrl: './signup.component.html',

  styleUrls: ['./signup.component.css']
```

```
import { Component, OnInit } from '@angular/core';

import {FormBuilder, FormGroup, Validators } from '@angular/forms';
```

```
@Component({
  selector: 'app-signup',
  templateUrl: './signup.component.html',
  styleUrls: ['./signup.component.css']
})
export class SignupComponent implements OnInit {
  submitted = false;
  registerForm: FormGroup;
  constructor(private formBuilder: FormBuilder) { }
  ngOnInit() {
        this.registerForm = this.formBuilder.group({
        firstName: ['', Validators.required],
        lastName: ['', Validators.required],
        email: ['', [Validators.required, Validators.email]],
        password: ['', [Validators.required, Validators.minLength(6)]]
        });
}
get f() { return this.registerForm.controls; }
onSubmit() {
  this.submitted = true;

  // stop here if form is invalid
  if (this.registerForm.invalid) {
        return;
  }

  alert('Your request has been submitted for approval')

}
```